A distributed accelerated gradient algorithm for distributed model predictive control of a hydro power valley

Minh Dang Doan[*] **Pontus Giselsson**[**] **Tamás Keviczky**[***]
Bart De Schutter[***] **Anders Rantzer**[**]

[*] *Department of Electrical, Electronic, and Telecommunications Engineering, Cantho University of Technology, Vietnam*
(email: minhdang@doan.vn)
[**] *Department of Automatic Control, Lund University, Sweden*
(email: {pontusg,rantzer}@control.lth.se)
[***] *Delft Center for Systems and Control, Delft University of Technology, The Netherlands*
(email: {t.keviczky,b.deschutter}@tudelft.nl)

Abstract

A distributed model predictive control (DMPC) approach based on distributed optimization is applied to the power reference tracking problem of a hydro power valley (HPV) system. The applied optimization algorithm is based on accelerated gradient methods and achieves a convergence rate of $O\left(\frac{1}{k^2}\right)$, where k is the iteration number. Major challenges in the control of the HPV include a nonlinear and large-scale model, nonsmoothness in the power-production functions, and a globally coupled cost function that prevents distributed schemes to be applied directly. We propose a linearization and approximation approach that accommodates the proposed the DMPC framework and provides very similar performance compared to a centralized solution in simulations. The provided numerical studies also suggest that for the sparsely interconnected system at hand, the distributed algorithm we propose is faster than a centralized state-of-the-art solver such as CPLEX.

Keywords: Hydro power control, Distributed optimization, Accelerated gradient algorithm, Model predictive control, Distributed model predictive control

1. Introduction

Hydro power plants generate electricity from potential energy and kinetic energy of natural water, and often a number of power plants are placed along a long river or a water body system to generate the power at different stages. Currently, hydro power is one of the most important means of renewable power generation in the world [36]. In order to meet the world's electricity demand, hydro power production should continue to grow due to the increasing cost of fossil fuels. However, hydro electricity, like any renewable energy, depends on the availability of a primary resource, in this case: water. The expected trend for future use of hydro power is to build small-scale plants that can generate electricity for a single community. Thus, an increasingly important objective of hydro power plants is to manage the available water resources efficiently, while following an optimal production profile with respect to changes in the electricity market, to maximize the long-term benefit of the plant. This water resource management must be compatible with ship navigation and irrigation, and it must respect environmental and safety constraints on levels and flow rates in the lakes and the rivers. By significantly increasing the power efficiency of hydro power valley (HPV) systems, real-time control of water flows becomes an important ingredient in achieving this objective.

An HPV may contain several rivers and lakes, spanning a wide geographical area and exhibiting complex dynamics. In order to tackle the plant-wide control of such a complex system, an HPV is often treated as a large-scale system consisting

of interacting subsystems. Large-scale system control has been an active research area that has resulted in a variety of control techniques, which can be classified in three main categories: decentralized control, distributed control, and centralized control. The application of these approaches can be found in a rich literature on control of water canals for irrigation and hydro systems [16, 14]. We are interested in applying model predictive control (MPC), a control method that has been successfully used in industry [25], thanks to its capability of handling hard constraints and the simple way of incorporating an economical objective by means of an optimization problem. For the control problem of open water systems, centralized MPC has been studied in numerical examples using nonlinear MPC approaches in combination with model smoothing and/or model reduction techniques [13, 18], and in real implementations with linear MPC of low-dimensional systems [33, 34]. However, centralized MPC has a drawback when controlling large-scale systems due to limitations in communications and the computational burden. These issues fostered the studies of decentralized MPC and distributed MPC for large-scale water systems. Early decentralized MPC methods for irrigation canals used the decomposition-coordination approach to obtain decentralized versions of LQ control [6]. Several decentralized MPC simulations applied to irrigation canals and rivers were presented in [7, 30, 10, 26]. Distributed MPC approaches based on coordination and cooperation for water delivery canals were presented in [7, 20, 12, 1]. The typical control objective in these studies is to regulate water levels and to deliver the required amount of water to the right place at some time in the future, i.e., the cost function does not have any special term except the quadratic penalties on the states and the inputs. On the other hand, in hydro power control, there are output penalty terms in the cost function that rep-

resent the objective of manipulating power production. Recent literature taking into account this cost function includes centralized nonlinear MPC with a parallel version of the multiple-shooting method for the optimal control problem using continuous nonlinear dynamics [29], and a software framework that formulates a discrete-time linear MPC controller with the possibility to integrate a nonlinear prediction model and to use commercial solvers to solve the optimization problem [24]. The hydro power control problem considered in the current paper is similar to the setup in [29, 24]. However, it distinguishes itself by using a distributed control structure that aims to avoid global communications and that divides the computational tasks into local sub-tasks that are handled by subsystems, making the approach more suitable for scaling up to even more complicated hydro power plants.

The distributed MPC design approach proposed in this paper is enabled by a distributed optimization algorithm that has recently been developed by the authors in [8]. This optimization algorithm is designed for a class of strongly convex problems with mixed 1-norm and 2-norm terms in the cost function, which perfectly suits the power reference tracking objective in the HPV control benchmark. The underlying optimization algorithm in [8], although being implemented in a distributed way, is proved to achieve the global optimum with an $O(\frac{1}{k^2})$ convergence rate, where k is the iteration number. This is a significant improvement compared to the distributed MPC methods presented in [5, 4, 9, 19], which achieve an $O(\frac{1}{k})$ convergence rate. There are three main challenges in applying distributed MPC using the algorithm from [8] to the HPV benchmark problem. The first one is that the nonlinear continuous-time model yields a relatively large linear model after spatial and temporal discretizations. We present a decentralized model order re-

duction method that significantly reduces the model complexity while maintaining prominent dynamics. The second challenge is that the power production functions are nonsmooth, which prevents gradient-based methods to be applied directly. A method to overcome this difficulty and to enable optimal control using the algorithm from [8] is also presented. The third challenge is that the whole system should follow a centralized power reference which, if the algorithm from [8] is applied directly, requires centralized communication. We propose a dynamic power division approach that allows to track this centralized power reference with only distributed communications. By means of numerical examples, we will demonstrate the fast convergence property of the distributed algorithm which, when implemented on a single core, can outperform a state-of-the-art centralized solver (CPLEX) when solving the same optimization problem.

The remaining parts of the paper are organized as follows. In Section 2, we describe the HPV system and the power reference tracking problem that were formulated in the HPV benchmark problem [28]. Section 3 provides a summary of the distributed optimization framework that the authors have developed in [8]. In Section 4, we present our approach for modeling and model reduction of the HPV system, followed by a reformulation of the MPC optimization problem, and developing a distributed estimator so that the closed loop distributed MPC scheme can be implemented using neighbor-to-neighbor communications only. The simulation results are presented in Section 5, which also features a comparison with centralized MPC and decentralized MPC. Through the various aspects of the comparison including performance, computational efficiency, and communication requirements, the advantages of the distributed MPC algorithm will be highlighted. Section 6 concludes the paper and outlines future work.

2. Problem description

In this section, we provide a summary of the hydro power valley benchmark [28] and we present the linearized model that serves as the starting point of our controller design.

2.1. Hydro power valley system

We consider a hydro power plant composed of several interconnected subsystems, as illustrated in Figure 1. The plant can be divided into 8 subsystems, of which subsystem S_1 is composed of the lakes L_1, L_2, the duct U_1 connecting them, and the ducts C_1, T_1 that connect L_1 with the reaches[1] R_1, R_2, respectively. Subsystem S_2 is composed of the lake L_3 and the ducts C_2, T_2 that connect L_3 to the reaches R_4, R_5, respectively. There are 6 other subsystems, each of which consists of a reach and a dam at the end of the reach. These six reaches R_1, R_2, R_3, R_4, R_5, and R_6 are connected in series, separated by the dams D_1, D_2, D_3, D_4, and D_5. The large lake that follows the dam D_6 is assumed to have a fixed water level, which will absorb all the discharge. The outside water flows enter the system at the upstream end of reach R_1 and at the middle of reach R_3.

There are structures placed in the ducts and at the dams to control the flows. These are the turbines placed in the ducts T_1, T_2 and at each dam for power production. In the ducts C_1, C_2 there are composite structures that can either function as pumps (for transporting water to the lakes) or as turbines (when water is drained from the lakes).

The whole system has 10 manipulated variables, which are composed of six dam flows (q_{D1}, q_{D2}, q_{D3}, q_{D4}, q_{D5}, q_{D6}), two turbine flows (q_{T1}, q_{T2}) and two

[1] A reach is a river segment between two dams.

pump/turbine flows (q_{C1}, q_{C2}). Further, the system has 9 measured variables, the water levels in the three lakes (h_{L1}, h_{L2}, h_{L3}) and the water levels at the end of each reach (h_{R1}, h_{R2}, h_{R3}, h_{R4}, h_{R5}, h_{R6}).

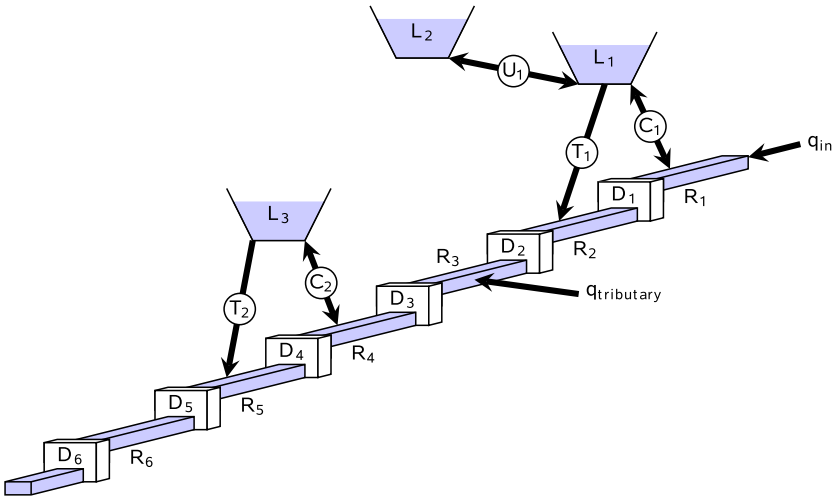


Figure 1: Overview of the HD-MPC hydro power valley system [28]

2.2. Power reference tracking problem

One of the control problems specified in [28] is the power reference tracking problem. We introduce state variables x, which consist of water levels in the lakes and reaches and water flows within the reaches, and control variables q, which are the manipulated water flows. The problem is to track a power production profile, $p^{\text{ref}}(t)$, on a daily basis using the following cost function:

$$J \triangleq \int_0^T \gamma \left| p^{\text{ref}}(t) - \sum_{i=1}^{8} p_i(x(t), q(t)) \right| dt + \sum_{i=1}^{8} \int_0^T (x_i(t) - x_i^{\text{ss}})^T Q_i (x_i(t) - x_i^{\text{ss}}) dt$$

$$+\sum_{i=1}^{8}\int_0^T (q_i(t)-q_i^{\text{ss}})^T R_i (q_i(t)-q_i^{\text{ss}})dt \tag{1}$$

subject to the nonlinear dynamics and linear constraints on outputs and inputs as specified in [28]. The weights $Q_i, R_i, i = 1,\dots,8$, γ, and the testing period T are parameters of the benchmark.

The quadratic term in the cost function represents the penalties on the state deviation from the steady state x^{ss} and the energy used for manipulating the inputs away from the steady state flows q^{ss}. The 1-norm term represents the power reference tracking mismatch, in which the function p^{ref} is the power reference and the function p_i represents the locally produced/consumed power by a subsystem $i \in \{1,\dots,8\}$. For $i = 1, 2$ the produced/consumed power is (cf. [28])

$$p_i(x(t),q(t)) = k_{C_i}(q_{C_i}(t))q_{C_i}(t)\Delta x_{C_i}(t) + k_{T_i}q_{T_i}(t)\Delta x_{T_i}(t) \tag{2}$$

where q_{C_i} and q_{T_i} are the flows through ducts C_i and T_i, Δx_{C_i} and Δx_{T_i} are the relative differences in water levels before and after ducts C_i and T_i respectively, k_{T_i} is the power coefficient of the turbine T_i, and

$$k_{C_i}(q_{C_i}(t)) = \begin{cases} k_{T_{C_i}}, & q_{C_i}(t) \geq 0 \\ k_{P_{C_i}}, & q_{C_i}(t) < 0 \end{cases} \tag{3}$$

is a discontinuous power coefficient that depends on whether the duct C_i acts as a turbine ($q_{C_i}(t) \geq 0$) or as a pump ($q_{C_i}(t) < 0$). For $i = 3,\dots,8$ we have

$$p_i(x(t),q(t)) = k_{D_{i-2}}q_{D_{i-2}}(t)\Delta x_{D_{i-2}}(t) \tag{4}$$

which is the power produced by the turbine located at dam D_{i-2}. The produced/consumed power functions given in (2) and (4) are nonlinear, and even nonsmooth for subsystems 1 and 2 due to the differences of $k_{T_{C_i}}$ and $k_{P_{C_i}}$ in (3), thus complicating a direct application of a standard MPC scheme.

Still, the complexity of the system and control objective suggests an optimization based control strategy, such as MPC. Further, the distributed nature of the system makes it possible to consider distributed MPC techniques. However, the stated optimization problem (1) is a nonlinear continuous-time dynamic optimization problem, which in general is very hard to solve. In the next sections we will discuss the modeling of the hydro power valley that leads to a linearized model.

2.3. Nonlinear hydro power valley model

The model of the reaches is based on the one-dimensional Saint Venant partial differential equation, representing the mass and momentum balance (see [28] for details):

$$\begin{cases} \dfrac{\partial q(t,z)}{\partial z} + \dfrac{\partial s(t,z)}{\partial t} = 0 \\ \dfrac{1}{g}\dfrac{\partial}{\partial t}\left(\dfrac{q(t,z)}{s(t,z)}\right) + \dfrac{1}{2g}\dfrac{\partial}{\partial z}\left(\dfrac{q^2(t,z)}{s^2(t,z)}\right) + \dfrac{\partial h(t,z)}{\partial z} + I_{\mathrm{f}}(t,z) - I_0(z) = 0 \end{cases} \tag{5}$$

with z the spatial variable, t the time variable, q the river flow (or discharge), s the cross-section surface of the river, h the water level w.r.t. the river bed, I_{f} the friction slope, $I_0(z)$ the river bed slope, and g the gravitational acceleration constant.

The partial differential equation (5) is converted into a system of ordinary differential equations by using spatial discretization. To achieve this, each reach is divided into 20 cells, yielding 20 additional states, which are the water levels at the beginning of the cells. For details of the spatial discretization and the equations for the resulting nonlinear dynamical system the reader is referred to [28, Section 2.1.1]. The resulting nonlinear dynamical system has in total 249 states, 10 inputs, and 9 outputs.

2.4. Model linearization and discretization

As mentioned in Section 2.3 a set of nonlinear ordinary differential equations that describe the hydro power valley dynamics is presented in [28, Section 2.1.1]. A linear continuous-time model which is linearized around the steady state operating point $(x^{\text{ss}}, q^{\text{ss}})$ is also provided in the HPV benchmark package [28]. Discretizing this model using zero-order-hold gives a discrete-time linear system with 249 states and 10 inputs. The coupling of the subsystems is through the inputs only. This implies that discretization using zero-order-hold of the continuous-time system keeps the structure of the original system description. Thus, the resulting discrete time system has a block-diagonal dynamics matrix, a block-diagonal output matrix, and a sparse input matrix, and each subsystem $i = 1, \ldots, 8$ can be expressed in the following form:

$$\begin{aligned} x_i^{\text{d}}(k+1) &= A_{ii} x_i^{\text{d}}(k) + \sum_{j=1}^{8} B_{ij} q_j^{\text{d}}(k) \\ y_i^{\text{d}}(k) &= C_i x_i^{\text{d}}(k) \end{aligned} \tag{6}$$

in which the variables $x^{\text{d}}, q^{\text{d}}$, and y^{d} stand for the deviation from the steady-state values, and the subscripts i, j stand for the subsystem indices. As mentioned the subsystems are coupled through the inputs only and at least for some $j \in \{1, \ldots, 8\}$ we have $B_{ij} = 0$ for every $i = 1, \ldots, 8$.

The use of a discrete-time linearized model enables controller design with some specific approaches, which include our proposed distributed optimization technique presented in [8]. Before describing our main contributions, we now provide a summary of this distributed optimization framework in the next section.

3. Distributed optimization framework for MPC

In this section, we describe the distributed optimization algorithm developed in [8] which is based on an accelerated gradient method. The first accelerated gradient method was developed in [21] and further elaborated and extended in [2, 22, 23, 31, 32]. The main idea of the algorithm presented in [8] is to exploit the problem structure of the dual problem such that accelerated gradient computations can be distributed to subsystems. Hence, the distributed algorithm effectively solves the centralized optimization problem. Dual decomposition has been used in the past to tackle the complexity of large-scale optimization problems arising in water supply networks [3]. In our work however, in addition to simplifying the local computations, we apply this decomposition philosophy in order to distribute the decision-making process.

The algorithm in [8] is developed to handle optimization problems of the form

$$\begin{aligned} \min_{\mathbf{x},\mathbf{x}_\mathrm{a}} \ & \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + g^T\mathbf{x} + \gamma\|\mathbf{x}_\mathrm{a}\|_1 \\ \text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{C}\mathbf{x} \leq \mathbf{d} \\ & \mathbf{x}_\mathrm{a} = \mathbf{P}\mathbf{x} - \mathbf{p} \end{aligned} \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{x}_\mathrm{a} \in \mathbb{R}^m$ are vectors of decision variables, and $\mathbf{x}$ is partitioned according to:

$$\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_M^T]^T, \tag{8}$$

and $\mathbf{x}_i \in \mathbb{R}^{n_i}$. Further, the matrix $\mathbf{H} \in \mathbb{R}^{n\times n}$ is positive definite and block-diagonal, the matrices $\mathbf{A} \in \mathbb{R}^{q\times n}$, $\mathbf{C} \in \mathbb{R}^{r\times n}$, and $\mathbf{P} \in \mathbb{R}^{m\times n}$ have sparse structures, and $g \in \mathbb{R}^n$, $\mathbf{p} \in \mathbb{R}^m$, $\mathbf{b} \in \mathbb{R}^q$, $\mathbf{d} \in \mathbb{R}^r$. We introduce the partitions $g =$

$[g_1^T, \ldots, g_M^T]^T$, $\mathbf{p} = [\mathbf{p}_1^T, \ldots, \mathbf{p}_M^T]^T$, $\mathbf{b} = [\mathbf{b}_1^T, \ldots, \mathbf{b}_M^T]^T$, $\mathbf{d} = [\mathbf{d}_1^T, \ldots, \mathbf{d}_M^T]^T$,

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_M \end{bmatrix}, \qquad \mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \ldots & \mathbf{A}_{1M} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{M1} & \ldots & \mathbf{A}_{MM} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \ldots & \mathbf{C}_{1M} \\ \vdots & \ddots & \vdots \\ \mathbf{C}_{M1} & \ldots & \mathbf{C}_{MM} \end{bmatrix}, \qquad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{11} & \ldots & \mathbf{P}_{1M} \\ \vdots & \ddots & \vdots \\ \mathbf{P}_{M1} & \ldots & \mathbf{P}_{MM} \end{bmatrix}$$

where the partitions are introduced in accordance with (8) and $g_i \in \mathbb{R}^{n_i}$, $\mathbf{p}_i \in \mathbb{R}^{m_i}$, $\mathbf{b}_i \in \mathbb{R}^{q_i}$, $\mathbf{d}_i \in \mathbb{R}^{r_i}$, $H_i \in \mathbb{R}^{n_i \times n_i}$, $\mathbf{A}_{ij} \in \mathbb{R}^{q_i \times n_j}$, $\mathbf{C}_{ij} \in \mathbb{R}^{r_i \times n_j}$ and $\mathbf{P}_{ij} \in \mathbb{R}^{m_i \times n_j}$. The assumption on sparsity of $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{P}$ is that $\mathbf{A}_{ij} = 0$, $\mathbf{C}_{ij} = 0$, and $\mathbf{P}_{ij} = 0$ for some i, j and we assume that the constraint matrices are built such that $\mathbf{A}_{ii} \neq 0$, $\mathbf{C}_{ii} \neq 0$, and $\mathbf{P}_{ii} \neq 0$ for all $i = 1, \ldots, M$. Based on the coupling, we define for each subsystem a neighborhood set, denoted by $\mathcal{N}_i$, as follows:

$$\mathcal{N}_i = \big\{ j \in \{1, \ldots, M\} |\ \mathbf{A}_{ij} \neq 0 \text{ or } \mathbf{A}_{ji} \neq 0 \text{ or } \mathbf{C}_{ij} \neq 0 \text{ or } \mathbf{C}_{ji} \neq 0 \text{ or } \mathbf{P}_{ij} \neq 0 \text{ or } \mathbf{P}_{ji} \neq 0 \big\}. \tag{9}$$

Note that there are two type of equality constriants in (7), the first one involves only $\mathbf{x}$ and the matrix $\mathbf{A}$ has a sparsity pattern, i.e., there is no global coupling introduced in that equality constraint; the last one involves both $\mathbf{x}$ and $\mathbf{x}_\mathrm{a}$, moreover introduces a global coupling due to the fact that $\mathbf{x}_\mathrm{a}$ is penalized in the 1-norm term of the cost function, thus it is not straightforward to deal with this constraint as we could treat the first constraint. Throughout the paper, the dual variables corresponding to these constraints are treated differently, and a distributed approximation of the 1-norm term is introduced to treat the second type of equality constraint.

We introduce dual variables $\lambda \in \mathbb{R}^q, \mu \in \mathbb{R}^r, \nu \in \mathbb{R}^m$ for the equality constraints, inequality constraints, and equality constraints originating from the 1-norm cost in (7) respectively. We also introduce the dual variable partitions $\lambda = [\lambda_1^T, \ldots, \lambda_M^T]^T$, $\mu = [\mu_1^T, \ldots, \mu_M^T]^T$, and $\nu = [\nu_1^T, \ldots, \nu_M^T]^T$ where $\lambda_i \in \mathbb{R}^{q_i}$, $\mu_i \in \mathbb{R}^{r_i}$, and $\nu_i \in \mathbb{R}^{m_i}$. Based on [8], the dual problem of (7) can be cast as the minimization of the negative dual function

$$f(\lambda, \mu, \nu) = \frac{1}{2}(\mathbf{A}^T\lambda + \mathbf{C}^T\mu + P^T\nu)^T\mathbf{H}^{-1}(\mathbf{A}^T\lambda + \mathbf{C}^T\mu + \mathbf{P}^T\nu) + \\ + \mathbf{b}^T\lambda + \mathbf{d}^T\mu + \mathbf{p}^T\nu \tag{10}$$

and the dual variables are constrained to satisfy

$$\lambda \in \mathbb{R}^q, \qquad \mu \in \mathbb{R}^r_+, \qquad \nu \in [-\gamma, \gamma]^m \tag{11}$$

where $\mathbb{R}_+$ denotes the non-negative real orthant. The negative dual function (10) has a Lipschitz continuous gradient with constant (cf. [8])

$$L = \|[\mathbf{A}^T\ \mathbf{C}^T\ P^T]^T\mathbf{H}^{-1}[\mathbf{A}^T\ \mathbf{C}^T\ \mathbf{P}^T]\|_2 \tag{12}$$

and can hence be minimized using accelerated gradient methods. The distributed accelerated gradient method as presented in [8] is summarized below in a slightly different form that is adapted to our HPV application problem at hand.

Algorithm 1. ***Distributed accelerated gradient algorithm***

Initialize $\lambda^0 = \lambda^{-1}$, $\mu^0 = \mu^{-1}$, $\nu^0 = \nu^{-1}$ and $\mathbf{x}^{-1}$ with the last values from the previous sampling step. For the first sampling step, these variables are initialized by zeros.

In every node, i, the following computations are performed:

For $k = 0, 1, 2, \ldots$

1. *Compute*

$$\mathbf{x}_i^k = -\mathbf{H}_i^{-1}\Big(\sum_{j\in\mathcal{N}_i}\big(\mathbf{A}_{ji}^T\lambda_j + \mathbf{C}_{ji}^T\mu_j + \mathbf{P}_{ji}^T\nu_j\big)\Big)$$
$$\bar{\mathbf{x}}_i^k = \mathbf{x}_i^k + \frac{k-1}{k+2}(\mathbf{x}_i^k - \mathbf{x}_i^{k-1})$$

2. *Send $\bar{\mathbf{x}}_i^k$ to each $j \in \mathcal{N}_i$, receive $\bar{\mathbf{x}}_j^k$ from each $j \in \mathcal{N}_i$*

3. *Compute*

$$\lambda_i^{k+1} = \lambda_i^k + \frac{k-1}{k+2}(\lambda_i^k - \lambda_i^{k-1}) + \frac{1}{L}\Big(\sum_{j\in\mathcal{N}_i}\mathbf{A}_{ij}\bar{\mathbf{x}}_j - \mathbf{b}_i\Big)$$
$$\mu_i^{k+1} = \max\left\{0, \mu_i^k + \frac{k-1}{k+2}(\mu_i^k - \mu_i^{k-1}) + \frac{1}{L}\Big(\sum_{j\in\mathcal{N}_i}\mathbf{C}_{ij}\bar{\mathbf{x}}_j - \mathbf{d}_i\Big)\right\}$$
$$\nu_i^{k+1} = \min\left\{\gamma, \max\left[-\gamma, \nu_i^k + \frac{k-1}{k+2}(\nu_i^k - \nu_i^{k-1}) + \right.\right.$$
$$\left.\left. + \frac{1}{L}\Big(\sum_{j\in\mathcal{N}_i}\mathbf{P}_{ij}\bar{\mathbf{x}}_j - \mathbf{p}_i\Big)\right]\right\}$$

4. *Send λ_i^{k+1}, μ_i^{k+1}, ν_i^{k+1} to each $j \in \mathcal{N}_i$, receive λ_j^{k+1}, μ_j^{k+1}, ν_j^{k+1} from each $j \in \mathcal{N}_i$.*

The Lipschitz constant L of ∇f is used in the algorithm. For MPC purposes we only need to compute L once in a centralized way and use it through all MPC problem instances.

Besides the suitability for distributed implementation, another merit of Algorithm 1 is its fast convergence rate. The main convergence results of Algorithm 1 are given in [8], stating that both the dual function value and the primal variables converge towards their respective optima with the rate of $O\left(\frac{1}{k^2}\right)$ where k is the iteration index. This convergence rate is much better than the convergence rate of classical gradient-based optimization algorithms, which is $O\left(\frac{1}{k}\right)$.

4. Control of HPV using distributed MPC

We have so far described the linear discrete-time model of the HPV in Section 2 and the fast distributed optimization method, Algorithm 1, that serves as a basis for designing a distributed model predictive controller to be applied to the HPV. However, there are three major challenges for this application. First, the linear discrete-time model cannot be directly used in an MPC context due to the existence of a number of unobservable and uncontrollable modes. These unobservable/uncontrollable modes are a result of the spatial discretization in each reach which creates states that cannot be observed/controlled separately. In addition, the linear discrete-time model has a large number of states, causing a large computational burden. Second, the power functions associated with the ducts C_1 and C_2 are nonsmooth (cf. (2) and (3)). The nonsmoothness is caused by the fact that the flow through C_1 and C_2 is bidirectional and the powers consumed/produced do not have equivalent coefficients. The third major challenge is the global coupling in the cost function due to the fact that we have to track a central power reference function that specifies the desired sum of locally generated power outputs. This global coupling prevents a distributed implementation of Algorithm 1 since the sparsity in the constraints is lost. These issues are addressed in the following sections.

4.1. Modification of the linear model

In this section we show how to create a model of the HPV that is suitable for the DMPC framework presented in [8]. First we present a model reduction technique that keeps the system structure, then the nonsmooth power function is treated.

4.1.1. Decentralized model order reduction

The block-diagonal structure of discrete-time dynamical system (6) makes it possible to perform model reduction on each subsystem individually. Several model reduction methods have been proposed for interconnected systems [35, 27]. In this work, we use a straightforward balanced truncation method [11, 17] to reduce the order of each local model (6).

Let us introduce $B_i = [B_{i1} \dots B_{i8}]$ and $q^{\mathrm{d}} = [(q_1^{\mathrm{d}})^T \dots (q_8^{\mathrm{d}})^T]^T$ to get the following discrete-time linear model of each subsystem:

$$\begin{aligned} x_i^{\mathrm{d}}(k+1) &= A_{ii} x_i^{\mathrm{d}}(k) + B_i q^{\mathrm{d}}(k) \\ y_i^{\mathrm{d}}(k) &= C_i x_i^{\mathrm{d}}(k). \end{aligned} \tag{13}$$

Applying the balanced truncation technique yields transformation matrices denoted by T_i^{r} and $T_i^{\mathrm{r,inv}}$ for each subsystem, where $T_i^{\mathrm{r}} T_i^{\mathrm{r,inv}} = I$. By denoting the new state variables, $x_i^{\mathrm{r}} = T_i^r x_i^{\mathrm{d}}$, and the control variable $q^{\mathrm{r}} = q^{\mathrm{d}}$, we represent the reduced order model as:

$$x_i^{\mathrm{r}}(k+1) = A_{ii}^{\mathrm{r}} x_i^{\mathrm{r}}(k) + B_i^{\mathrm{r}} q^{\mathrm{r}}(k) \tag{14}$$

$$y_i^{\mathrm{r}}(k) = C_i^{\mathrm{r}} x_i^{\mathrm{r}}(k) \tag{15}$$

where $A_{ii}^{\mathrm{r}} = T_i^{\mathrm{r}} A_{ii} T_i^{\mathrm{r,inv}}$, $B_i^{\mathrm{r}} = T_i^{\mathrm{r}} B_i$ and $C_i^{\mathrm{r}} = C_i T_i^{\mathrm{r,inv}}$. It should be noted that the block-sparsity structure of B_i^r is the same as in the non-reduced input matrix B_i, since the model reduction is performed for each local model separately. Moreover, all the modes of the reduced model are both observable and controllable.

The model reduction gives a 32-state reduced model that approximately represents the dynamics of the full linear model with 249 states.

4.1.2. Treatment of nonlinear and nonsmooth power function

One of the difficulties in applying a linear MPC approach to the hydro power valley is the nonsmoothness of the power function associated with the ducts C_1 and C_2, which is included in the expression for power generation (2) in subsystem 1 and subsystem 2, respectively. In order to handle this nonsmoothness, we use a double-flow technique, which means introducing two nonnegative positive variables to express the flow in $C_i, i = 1, 2$ at a sampling step k:

- $q_{C_{i\mathrm{P}}}(k)$: virtual flow such that C_i functions as a pump
- $q_{C_{i\mathrm{T}}}(k)$: virtual flow such that C_i functions as a turbine

The introduction of virtual flows requires the input-matrices, B_i^{r}, to be augmented with two extra columns identical to the ones multiplying $q_{C_i}, i = 1, 2$ with the opposite sign to capture that pump action is also introduced with a positive flow. The resulting reduced order model has 12 inputs instead of the original 10. Using the introduced flows $q_{C_{i\mathrm{P}}}$ and $q_{C_{i\mathrm{T}}}$, the power function (2) for subsystems 1 and 2 can be rewritten as

$$p_i(x(k), q(k)) = \left(k_{T_{C_i}} q_{C_{i\mathrm{T}}}(k) - k_{P_{C_i}} q_{C_{i\mathrm{P}}}(k)\right) \Delta x_{C_i}(k) + k_{T_i} q_{T_i}(k) \Delta x_{T_i}(k) \tag{16}$$

with the additional constraints that $q_{C_{i\mathrm{T}}}(k) \geq 0, q_{C_{i\mathrm{P}}}(k) \geq 0$ and $q_{C_{i\mathrm{T}}}(k) q_{C_{i\mathrm{P}}}(k) = 0$. The last constraint expresses the fact that water flows in only one direction at a time, i.e., that either the pump or the turbine is active. The resulting nonlinear expression (16) can in turn be linearized around the steady-state solution $(x^{\mathrm{ss}}, q^{\mathrm{ss}})$. Since $q_{C_i}^{\mathrm{ss}} = 0$ for $i = 1, 2$ we get the following linear local power production/consumption approximation for subsystems $i = 1, 2$:

$$\hat{p}_i(x(k), q(k)) = \Delta x_{C_i}^{\text{ss}} \begin{bmatrix} k_{T_{C_i}} & -k_{P_{C_i}} \end{bmatrix} \begin{bmatrix} q_{C_{i\text{T}}}(k) \\ q_{C_{i\text{P}}}(k) \end{bmatrix} + \\ + k_{T_i} q_{T_i}^{\text{ss}} \left(\Delta x_{T_i}(k) - \Delta x_{T_i}^{\text{ss}} \right) + k_{T_i} \Delta x_{T_i}^{\text{ss}} \left(q_{T_i}(k) - q_{T_i}^{\text{ss}} \right) + \\ + k_{T_i} q_{T_i}^{\text{ss}} \Delta x_{T_i}^{\text{ss}}$$

This reformulation results in a linear expression with a nonlinear constraint at each time step k, $q_{C_{i\text{T}}}(k) q_{C_{i\text{P}}}(k) = 0$, that approximates the original nonsmooth nonlinear power production/consumption expression (2). We show our approach to handle the nonlinear constraint in Section 4.2.

For subsystems $i = 3, \ldots, 8$ we have smooth power production expressions (4) that can be directly linearized without introducing virtual flows:

$$\hat{p}_i(x(k), q(k)) = k_{D_i} q_{D_i}^{\text{ss}} \Delta x_{D_i}^{\text{ss}} + k_{D_i} q_{D_i}^{\text{ss}} \left(\Delta x_{D_i}(k) - \Delta x_{D_i}^{\text{ss}} \right) + \\ + k_{D_i} \Delta x_{D_i}^{\text{ss}} \left(q_{D_i}(k) - q_{D_i}^{\text{ss}} \right)$$

4.2. HPV optimization problem formulation

In this section we formulate an optimization problem of the form (7) that can be used for power reference tracking in the HPV benchmark using MPC. We have obtained a linear discrete-time dynamical system (14)-(15) for the HPV with state variables x^{r} and control variables q^{r}. The constraints are upper and lower bounds on the outputs and inputs and their values can be found in [28]. Using the transformations matrices T_i^{r} and $T_i^{\text{r,inv}}$, these constraints can readily be recast as linear constraints for the reduced order model variables $x^{\text{r}}, q^{\text{r}}$. The power reference problem formulation (1) specifies a quadratic cost on states and control variables and a 1-norm penalty on deviations from the provided power reference, p^{ref}. For

control horizon, N, this optimization problem can be written as

$$\min_{\mathbf{x},\mathbf{x}_\mathrm{a}} \sum_{t=0}^{N-1} \left\{ \sum_{i=1}^{8} \left[x_i^\mathrm{r}(k)^T Q_i x_i^\mathrm{r}(k) + q_i^\mathrm{r}(k)^T R_i q_i^\mathrm{r}(k) \right] + \gamma \|x_\mathrm{a}(k)\|_1 \right\} \tag{17}$$

$$\begin{aligned}
\text{s.t.} \quad & (14), (15) && k = 0, \ldots, N-1 \quad i = 1, \ldots, 8 \\
& C_i^\mathrm{r} x_i^\mathrm{r}(k) \in \mathcal{Y}_i && k = 0, \ldots, N-1 \quad i = 1, \ldots, 8 \\
& q_i(k) \in \mathcal{Q}_i && k = 0, \ldots, N-1 \quad i = 1, \ldots, 8 \\
& x_\mathrm{a}(k) = p^\mathrm{ref}(k) - \textstyle\sum_{i=1}^{8} \hat{p}_i(x^\mathrm{r}(k), q^\mathrm{r}(k)) && k = 0, \ldots, N-1 \\
& q_{C_{i\mathrm{T}}}(k) q_{C_{i\mathrm{P}}}(k) = 0 && k = 0, \ldots, N-1 \quad i = 1, \ldots, 2
\end{aligned}$$

where $\mathcal{Y}_i$ and $\mathcal{Q}_i$ are sets representing the local output and input constraints, the additional variable x_a captures the power reference tracking mismatch, and the notation $\mathbf{x}$ represents the stack of variables $x_i^\mathrm{r}(k)$ and $q_i^\mathrm{r}(k)$ for all i and k, while $\mathbf{x}_\mathrm{a}$ is the stacked variable of $x_\mathrm{a}(k)$ for all k. Note that we can write $\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_8^T]^T$ where each $\mathbf{x}_i, i = 1, \ldots, 8$ includes all the variables that belong to subsystem i.

4.2.1. Power reference division

Since the original cost function contains a non-separable 1-norm term, the power reference constraints in the optimization problem (17) are coupled between all subsystems. This implies that Algorithm 1 requires some global communication even though the only information needed to be sent to the global coordinator is $\bar{p}_i(x^\mathrm{r}(k), q^\mathrm{r}(k))$ for $k = 0, \ldots, N-1$ from each subsystem $i = 1, \ldots, 8$.

In order to obtain a suitable dual problem, we first need to reformulate the cost function in a separable form. For the sake of brevity, we focus on one sampling step and drop the time index k. Thus for now our simplified objective is to decompose the following problem:

$$\min_{\{\mathbf{x}_i\}_{i=1,\ldots,8}} \quad \left| p^\mathrm{ref} - \sum_{i=1}^{8} P_i \mathbf{x} \right| \tag{18}$$

with $\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_8^T]^T$, and P_i the matrix coefficient such that the power function produced or consumed by each subsystem $\hat{p}_i(x^{\mathrm{r}}(k), q^{\mathrm{r}}(k))$ is linearized as $P_i\mathbf{x}(k)$.

In this section we present two different ways that avoid global communication when solving this problem. In the first approach, we divide and distribute the global power reference to the subsystems in a static fashion. In the second approach, we show how the subsystems can trade local power references between neighbors to achieve a satisfactory centralized reference tracking.

Static local power references. The idea here is straightforward. We divide the global power reference into local ones, i.e., p^{ref} is divided into local parts p_i^{ref}, $i = 1, \ldots, 8$. We have chosen to compute p_i^{ref} such that it satisfies

$$\frac{p_i^{\mathrm{ref}}(k)}{\sum_{i=1}^{8} p_i^{\mathrm{ref}}(k)} = \frac{p_i(x^{\mathrm{ss}}, q^{\mathrm{ss}})}{\sum_{i=1}^{8} p_i(x^{\mathrm{ss}}, q^{\mathrm{ss}})}, \quad \text{for } k = 0, \ldots, N-1 \tag{19}$$

with $p_i(x^{\mathrm{ss}}, q^{\mathrm{ss}})$ the power produced by subsystem i in the steady-state condition.

This means that the fraction of the total power reference given to subsystem i is constant. The optimization problem is changed accordingly, i.e., the following cost function can be used instead of (18):

$$\min_{\{\mathbf{x}_i\}_{i=1,\ldots,8}} \quad \sum_{i=1}^{8} \left| p_i^{\mathrm{ref}} - P_i\mathbf{x} \right| \tag{20}$$

with $\mathbf{x} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_8^T]^T$. This allows for a distributed implementation since the matrix P_i introduces only local couplings, i.e., subsystem i needs only neighboring and local water levels and local water flows to compute the corresponding power output. The disadvantage of the static power reference division is that the global power reference tracking is not very accurate, as will be shown in the simulations section.

Dynamic local power references. The static power division essentially means that each subsystem always tracks a fraction of power reference that is equal to the proportion it produces in the steady-state condition. When the total power reference deviates significantly from the steady-state power, this idea may not work well since the proportional change of the local power reference can lead to sub-optimal performance. Inspired by an idea in [15], we now introduce the dynamic power division, in which the subsystems have more flexibility in choosing the appropriate local power reference to be tracked. The main idea is that each subsystem will exchange power references with its direct neighbors.

Let us define for each pair (i, j) with $j \in \mathcal{N}_i$ a node that is in charge of determining the power exchange variable between subsystems i and j, denoted by δ_{ij} if node i is in charge and by δ_{ji} if node j is in charge [2]. Then for each subsystem we form the set [3]:

$$\Delta_i = \{j \mid j \in \mathcal{N}_i, i \text{ is in charge of } \delta_{ij}\}. \tag{21}$$

Now we replace (18) by the following cost function:

$$\min_{\{\mathbf{x}_i, \delta_i\}_{i=1,\dots,8}} \sum_{i=1}^{8} \left| p_i^{\text{ref}} + \sum_{j \in \Delta_i} \delta_{ij} - \sum_{j \in \mathcal{N}_i \setminus \Delta_i} \delta_{ji} - P_i \mathbf{x} \right| \tag{22}$$

with δ_i the vector containing all $\delta_{ij}, j \in \Delta_i$, and p_i^{ref} the nominal power reference for subsystem i. In words, the local power reference for each subsystem i deviates from the nominal value by adding the exchange amounts of the links that i

[2] Note that here we discuss the power division for each sampling step, i.e., there are $\delta_{ij}(k)$ or $\delta_{ji}(k)$ with $k = 0, \dots, N-1$.

[3] A simple way is to let the subsystem with smaller index lead the exchange, i.e., $\Delta_i = \{j | j \in \mathcal{N}_i, j > i\}$.

manages and subtracting the exchange amounts of the links that affect i but are decided upon by its neighbors. Note that problem (22) has a sparse structure that complies with the existing sparse structure of the HPV system, i.e., this method does not expand the neighborhood set of each subsystem.

The advantage of this dynamic power division is that it makes use of the existing network topology to form a sparse cost function, and the total power reference is preserved even if the local power references can deviate from the nominal values, i.e., we always have:

$$\sum_{i=1}^{8} \left\{ p_i^{\text{ref}} + \sum_{j \in \Delta_i} \delta_{ij} - \sum_{j \in \mathcal{N}_i \setminus \Delta_i} \delta_{ji} \right\} = p^{\text{ref}} \tag{23}$$

Now that we have a separable cost function by using either a static or a dynamic power division technique, we can cast the approximate optimization problem in the form (7) that has a separable dual problem, and apply Algorithm 1 at every sampling step. However, due to the requirement of positive definiteness of the quadratic term in the objective function, the introduced power exchange variables δ_{ij} must be penalized with a positive definite quadratic term. This implies that power reference exchange has an associated cost.

Communication structures. In the preceding sections we have presented three different ways to handle the power reference term. The first is the one with centralized power reference term which we hereby denote by GLOBAL–REF. The second is the one with static local power references which we denote by LOC–REF–STAT. The third is the dynamic local power reference which from here on is denoted by LOC–REF–DYN. In Table 1 we provide an overview of the neighborhood sets $\mathcal{N}_i$ for the different power reference tracking schemes.

Table 1: Neighborhoods of subsystems ($\mathcal{N}_i$)

Subsystem	GLOBAL–REF	LOC–REF–DYN	LOC–REF-STAT
1	$\{1,\ldots,8\}$	$\{1,3,4\}$	$\{1,3,4\}$
2	$\{1,\ldots,8\}$	$\{2,6,7\}$	$\{2,6,7\}$
3	$\{1,\ldots,8\}$	$\{3,1,4\}$	$\{3,1,4\}$
4	$\{1,\ldots,8\}$	$\{4,1,3,5\}$	$\{4,1,3,5\}$
5	$\{1,\ldots,8\}$	$\{5,4,6\}$	$\{5,4,6\}$
6	$\{1,\ldots,8\}$	$\{6,2,7,5\}$	$\{6,2,7,5\}$
7	$\{1,\ldots,8\}$	$\{7,2,6,8\}$	$\{7,2,6,8\}$
8	$\{1,\ldots,8\}$	$\{8,7\}$	$\{8,7\}$

We can see that all subsystems have the same neighborhood sets for the dynamic local reference tracking and the static local reference tracking.

4.2.2. Relaxation of nonlinear constraint

The second issue that hinders the optimization problem (17) from being solved using Algorithm 1 are the nonlinear constraints $q_{C_{i\mathrm{T}}}(k)q_{C_{i\mathrm{P}}}(k) = 0$ with $i = 1, 2$. In this section we present a way to relax these constraints.

Assuming in the cost function we have the penalty $R_{C_i}[q_{C_{i\mathrm{T}}} q_{C_{i\mathrm{P}}}]^T$ on the pump and turbine action in ducts C_i, $i = 1, 2$, with

$$R_{C_i} = \begin{bmatrix} R_{C_{i\mathrm{T}}} & 0 \\ 0 & R_{C_{i\mathrm{P}}} \end{bmatrix}. \tag{24}$$

We also have the constraints that $q_{C_{i\mathrm{P}}}(k) \geq 0, q_{C_{i\mathrm{T}}}(k) \geq 0$ and $q_{C_{i\mathrm{T}}}(k)q_{C_{i\mathrm{P}}}(k) = 0$. We relax this by removing the nonlinear constraint and adding a cross-penalty $\alpha\sqrt{R_{C_{1\mathrm{P}}}R_{C_{1\mathrm{T}}}}$ for some $\alpha \in (0, 1)$ in the cost function, i.e., we set

$$R_{C_i} = \begin{bmatrix} R_{C_{i\mathrm{T}}} & \alpha\sqrt{R_{C_{i\mathrm{P}}}R_{C_{i\mathrm{T}}}} \\ \alpha\sqrt{R_{C_{i\mathrm{P}}}R_{C_{i\mathrm{T}}}} & R_{C_{i\mathrm{P}}} \end{bmatrix}. \tag{25}$$

This relaxation is implementable using the proposed algorithm since the nonlinear constraint is removed and replaced by a cross-penalty. The cross-penalty gives an additional cost if both $q_{C_{i\mathrm{T}}}$ and $q_{C_{i\mathrm{P}}}$ are non-zero. The closer α is to 1, the larger the penalty. For $\alpha \geq 1$ it is easily verified that we lose strong convexity on the quadratic cost function, i.e., R_{C_i} loses positive definiteness and such choices for α are therefore prohibited.

The relaxation is not equivalent to the original nonlinear constraint and thus cannot guarantee that the nonlinear constraint is respected using this relaxation. However, it turns out that the optimal solution using the cross-penalty in the cost (25) in most simulated cases coincides with the optimal solution when the nonlinear constraint $q_{C_{i\mathrm{T}}}(k)q_{C_{i\mathrm{P}}}(k) = 0$ and the original diagonal cost (24) are enforced. In some cases however, the optimal solution using the relaxation does not respect the nonlinear constraint. To address this, a two-phase optimization strategy is developed and presented next.

4.2.3. Two-phase optimization

We propose a two-phase optimization strategy as an ad-hoc branch and bound optimization routine that uses two consecutive optimizations. In the first optimization the relaxed optimization problem is solved. If the nonlinear constraints are respected, i.e., we get a solution that satisfies $q_{C_{i\mathrm{T}}}(k)q_{C_{i\mathrm{P}}}(k) = 0$, the global optimal solution for the non-relaxed problem is found. If some of the nonlinear constraints do not hold, the optimization routine is restarted with setting the smaller flow between $q_{C_{i\mathrm{T}}}(k)$ and $q_{C_{i\mathrm{P}}}(k)$ to zero, for $i = 1, 2$, $k = 0, \ldots, N-1$. The resulting algorithm is summarized below.

Algorithm 2. ***Distributed branch and bound algorithm***

1. *Solve the relaxed problem using Algorithm 1*
2. **If** $q_{C_{i\mathrm{T}}}(k)q_{C_{i\mathrm{P}}}(k) \neq 0$, $\qquad i = 1, 2, t = 0, \ldots, N-1$

   If $q_{C_{i\mathrm{T}}}(k) > q_{C_{i\mathrm{P}}}(k)$

   Add constraint: $q_{C_{i\mathrm{P}}}(k) = 0$

   Else

   Add constraint: $q_{C_{i\mathrm{T}}}(k) = 0$

   End

   End
3. *Solve relaxed problem using Algorithm 1 with the additional flow constraints*

This ad-hoc branch and bound technique does not theoretically guarantee that the optimal flow directions are chosen. However, we can guarantee that the non-linear constraints are always satisfied. Further, for the distributed MPC formulation we will see in the simulations section that the global optimal solution for the non-relaxed problem is found at every time step using this branch and bound algorithm.

4.3. Distributed estimation

From Section 2 we know that not all states can be measured, which implies that an observer needs to be used to feed an initial condition to the optimizer. The reduced-order linear model (14)-(15) has local dynamics and outputs only, which implies that an observer can be designed in decentralized fashion. We introduce the local estimate $\hat{x}_i^{\mathrm{r}}$ and the local observer-gain K_i, and the following local observer dynamics

$$\hat{x}_i^{\mathrm{r}}(k+1) = A_{ii}^{\mathrm{r}}\hat{x}_i^{\mathrm{r}}(k) + B_i^{\mathrm{r}}q^{\mathrm{r}}(k) + K_i(y_i^{\mathrm{r}}(k) - C_i^{\mathrm{r}}\hat{x}_i^{\mathrm{r}}(k))$$

Because of the sparse structure of B_i^{r} this observer can be implemented in a distributed fashion where only the inflows to subsystem i need to be communicated. The estimation error $\widetilde{x}_i^{\mathrm{r}} = x_i^{\mathrm{r}} - \hat{x}_i^{\mathrm{r}}$ has local error dynamics

$$\widetilde{x}_i^{\mathrm{r}}(k+1) = (A_{ii}^{\mathrm{r}} - K_i C_i^{\mathrm{r}})\widetilde{x}_i^{\mathrm{r}}(k)$$

Thus, the observer can be designed in a decentralized fashion and be implemented in a distributed fashion.

5. Simulation results

We perform distributed MPC simulations of the hydro power valley using 3 different ways of handling the power reference: GLOBAL–REF, LOC–REF–DYN, and LOC–REF–STAT, using the proposed Algorithm 2. We also solve the problem (17) using a state-of-the-art MIQP-solver, namely CPLEX. In CPLEX the nonlinear constraints given in (17) can be addressed by introducing binary variables. More specifically, for each duct $C_i, i = 1, 2$, we define two virtual flows, $q_{C_{i\mathrm{P}}}$ and $q_{C_{i\mathrm{T}}}$, and require that both values are nonnegative. Each virtual flow has a maximum capacity, hence the constraints for these flows are:

$$\begin{aligned} 0 \leq q_{C_{i\mathrm{P}}} &\leq q_{C_{i\mathrm{P}}}^{\max} \\ 0 \leq q_{C_{i\mathrm{T}}} &\leq q_{C_{i\mathrm{T}}}^{\max} \end{aligned} \tag{26}$$

We introduce binary variables $b_i \in \{0, 1\}$ and impose the following constraints:

$$\begin{aligned} q_{C_{i\mathrm{T}}} &\leq q_{C_{i\mathrm{T}}}^{\max} b_i \\ q_{C_{i\mathrm{P}}} &\leq q_{C_{i\mathrm{P}}}^{\max} (1 - b_i) \end{aligned} \tag{27}$$

The constraints (26) and (27) ensure that either $q_{C_{i\mathrm{P}}} = 0, q_{C_{i\mathrm{T}}} \geq 0$ (if $b_i = 1$) or $q_{C_{i\mathrm{T}}} = 0, q_{C_{i\mathrm{P}}} \geq 0$ (if $b_i = 0$).

This formulation results in an MIQP for which there are efficient Branch-and-Bound algorithms implemented in CPLEX. To make the 1-norm term in (17) fit the MIQP-formulation used in CPLEX we introduce auxiliary variables v and use the following equivalent reformulation

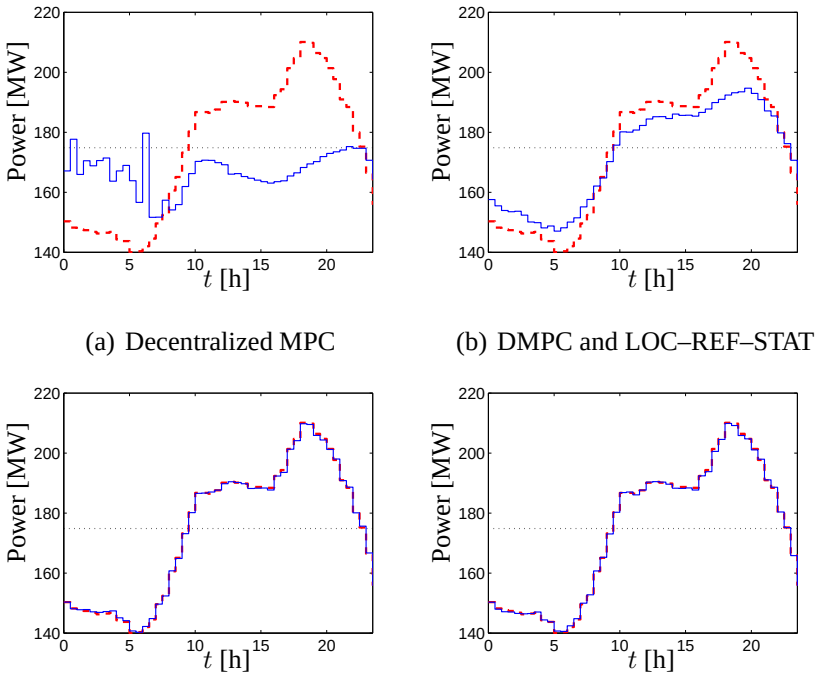
$$\min_x \|Px - p\|_1 \qquad \Leftrightarrow \qquad \min_{x,v} 1^T v$$
$$\text{s.t.} -v \leq Px - p \leq v$$

We also compare the proposed distributed MPC method to a decentralized MPC approach in which each subsystem solves its own local MPC problem without any communication, in order to show the advantage of DMPC w.r.t. decentralized MPC.

5.1. Simulation details

We use the original nonlinear continuous model presented in [28] as simulation model. The ode-solver *ode15s* in MATLAB is used to perform the simulations. A MATLAB function that computes the derivatives needed by *ode15s* is provided in the benchmark package [28]. The control system consists of the distributed observer from Section 4.3 which feeds Algorithm 2, with estimates of the current state.

Besides the mismatch between the model used for control and the model used for simulation we have also added bounded process noise to capture mismatch between the simulation model and the real plant. The magnitude of the worst case process noise was chosen to be 1% of the steady-state level x^{ss}. We also use bounded additive measurement noise where the measured water levels are within ± 3 cm from the actual water levels.



(a) Decentralized MPC (b) DMPC and LOC–REF–STAT

(c) DMPC and LOC–REF–DYN (d) DMPC and GLOBAL–REF

Figure 2: Comparison of power reference tracking performance using DMPC and decentralized MPC approaches. Solid lines: produced power, dashed lines: reference power, dotted lines: steady state power.

We use a sampling time of 30 minutes in all simulations and the control horizon is $N = 10$, i.e., 5 hours. The simulations are performed over a 24 hour period since the power reference trajectories are periodic with this interval.

All simulations and optimizations were implemented on a PC running MATLAB on Linux with an Intel(R) Core(TM) i7 CPU running at 3 GHz and with 4 GB RAM.

5.2. Control performance comparison

The power reference tracking results are plotted in Figures 2(d)–2(a) where the full power reference and the sum of the local power productions are plotted. The scheme GLOBAL–REF achieves very good tracking performance, while the scheme LOC–REF–STAT shows a significant deterioration in tracking performance. However, the introduction of the possibility to exchange power references in LOC–REF–DYN between subsystems restores the very good tracking performance while keeping the computations distributed. The tracking performance of the decentralized MPC approach is very poor, due to the lack of communications. Hence, it is recommended not to use a decentralized MPC approach, unless communication is prohibited due to the lack of communication facilities or due to the policy of different authorities.

In Appendix A and Appendix B there are figures that show the input and output evolutions and the corresponding constraints with the scheme LOC–REF–DYN. We can observe that all constraints are satisfied despite disturbances, model mismatch, and the use of an observer. For the schemes GLOBAL–REF and LOC–REF–STAT all the constraints on the inputs and outputs are also satisfied.

During the simulations, it is observed that all schemes achieve stable closed-loop behaviours, which can be explained that the HPV system is already marginally stable and does not have critical dynamics, and the prediction horizon is long enough so that the MPC controllers do not introduce instability to the closed loop. Note that neither the centralized MPC nor the distributed or decentralized MPC approaches used in this simulations employ a method that provides guaranteed stability to the closed-loop system, since this property is beyond the scope of this paper. Based on the techniques for distributing the computation and improving

the efficiency of the algorithm that are proposed in this paper, one can further incorporate other MPC schemes that guarantee the closed-loop stability, which could be important for other types of applications where there are large mismatch between the nonlinear and the linearized models.

5.3. Computational efficiency/accuracy

In Table 2 we provide a comparison of the execution times of the centralized MPC problems (17). We compare the distributed Algorithm 2 to the solver CPLEX when solving (17), i.e., with power-division GLOBAL–REF in Algorithm 2. To solve this problem using CPLEX, an MIQP formulation is used. In every iteration of Algorithm 2 the relaxed problem is solved twice. We also compare the above execution times to the case when we solve the first relaxed problem in Algorithm 2, which is a QP, using CPLEX. At each sampling step, the same problem is solved, and the execution time t is measured. Although in this example the solvers easily solve the problem within the time frame of the sampling time, we can see that the computation time for our MATLAB-implemented algorithm is always lower than the C-implemented CPLEX for both the MIQP and QP cases.

Table 2: Comparison of computation time between Algorithm 2 and CPLEX for 48 instance of the same problem

	Algorithm 2	CPLEX for MIQP	CPLEX for QP
min t (s)	0.023	0.087	0.049
max t (s)	0.086	0.121	0.089
average t (s)	0.054	0.098	0.063
std dev t (s)	0.017	0.009	0.009

As previously discussed, Algorithm 2 cannot guarantee that the global op-

timum for (17) is found. However, in the DMPC simulations presented in this section the global optimum of (17) is found at every sampling step using Algorithm 2.

5.4. Communication requirements

The sizes of the optimization problems using power reference division GLOBAL–REF, LOC–REF–DYN or LOC–REF–STAT are almost equal. Comparing GLOBAL–REF to LOC–REF–STAT we get some additional constraints due to the power reference division and comparing LOC–REF–DYN to LOC–REF–STAT we get some additional decision variables δ_{ij} to enable distributed power reference reassignment.

In Table 3 the number of iterations n_{iter} needed to obtain the solution is presented. The average and max values of n_{iter} and the standard deviation are computed using 48 simulation steps, i.e., 24 hours.

Table 3: Number of iterations to solve the MPC optimization in one step

	Alg. 1 with **GLOBAL–REF**	Alg. 1 with **LOC–REF–DYN**	Alg. 1 with **LOC–REF–STAT**
average n_{iter}	311.3	579.1	942.5
max n_{iter}	498	1054	2751
std dev n_{iter}	93.8	210.9	440.8

We can notice that different DMPC schemes converge with different average numbers of iterations. The reason is that for LOC–REF–STAT it is more difficult to satisfy the different 1-norm terms with equality, i.e., to follow the local power references. This implies that the corresponding dual variable ν becomes large (close or equal to γ) and it takes more iterations to achieve convergence. As

a result, the scheme LOC–REF–STAT with a simpler communication structure might require more communication resources than e.g., GLOBAL–REF, which has a more complicated communication structure but needs fewer iterations.

In order to estimate the total time required for communications within each sampling time, we now assume the worst case happens in every iteration of Algorithm 2, in which Algorithm 1 has to be executed two times. In Algorithm 1, also assume the worst case that every primal and dual variable has to be exchanged between distributed controllers, with prediction horizon $N = 10$ there are $10 \times (44 + 65) = 1090$ variables to be transmitted once per iteration. Let each variable be a 32-bit floating-point, then the total time it would take for transmitting exchanged variables in 1000 iterations is:

$$2 \times 1090 \times 32 \times 1000 = 69,760,000(\text{bits}) \tag{28}$$

or roughly 70 Mbits. With a decent wireless network that can connect each two nodes with a rated transfer as 7 Mbps, the total time for communications is less than 10 seconds for one thousand iterations. Note that in practice, there should be more communication delays due to the initialization of transmissions. Since the communication time is considerably shorter than the sampling time of 30 minutes, the iterative methods taking about one thousand iterations sampling time can still be implemented in real time.

The scheme LOC–REF–DYN performs very well in terms of communication, computation, as well as performance aspects and is therefore the chosen candidate for distributed implementation for the given case study.

6. Conclusions and future work

The proposed distributed MPC approach has been applied to the power reference tracking problem of the HD-MPC hydro power valley benchmark. Two distributed schemes have been compared to centralized and decentralized MPC methods. We have provided relaxations and approximations for the original nonlinear nonsmooth problem formulation as well as proposed a way to follow a centralized power reference in a distributed fashion. Furthermore, we have presented a practical branch-and-bound algorithm that solves all optimization problems encountered in the simulations and achieves as good performance as the centralized MPC that is known to have global optimum. The simulation results show that the introduced approximations and relaxations capture the behavior of the system well and that very good control performance is achieved. Finally, a comparison to state-of-the-art optimization software (CPLEX) shows that the proposed algorithm has significantly better execution times in general.

As the next step before implementation in real plants, the proposed distributed MPC approach should be tested against different hydraulic scenarios and other HPV setups. To cope with varying water flows entering the system, these should be estimated and compensated for. Furthermore, a weather model could be included that estimates the future inflows to the system.

7. Acknowledgments

The authors were supported by the European Union Seventh Framework STREP project “Hierarchical and distributed model predictive control (HD-MPC)” with contract number INFSO-ICT-223854, the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement no. 257462 HYCON2 Net-

work of Excellence, the BSIK project “Next Generation Infrastructures (NGI)”, and the Swedish Research Council through the Linnaeus center LCCC.

References

[1] Anand, A., Joshua, G., Sundaramoorthy, S., Samavedham, L., Apr. 2011. Coordinating multiple model predictive controllers for multi-reservoir management. In: Proceedings of the 2011 IEEE International Conference on Networking, Sensing and Control. Delft, The Netherlands, pp. 1–6.

[2] Beck, A., Teboulle, M., Oct 2009. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences 2 (1), 183–202.

[3] Carpentier, P., Cohen, G., Sep. 1993. Applied mathematics in water supply network management. Automatica 29 (5), 1215 – 1250.

[4] Doan, D., Keviczky, T., Necoara, I., Diehl, M., De Schutter, B., Sep. 2009. A distributed version of Han’s method for DMPC using local communications only. Control Engineering and Applied Informatics 11 (3), 6–15.

[5] Doan, M. D., Keviczky, T., De Schutter, B., 2011. An iterative scheme for distributed model predictive control using Fenchel’s duality. Journal of Process Control 21 (5), 746–755, Special Issue on Hierarchical and Distributed Model Predictive Control.

[6] Fawal, H., Georges, D., Bornard, G., Oct. 1998. Optimal control of complex irrigation systems via decomposition-coordination and the use of augmented

Lagrangian. In: Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics. Vol. 4. San Diego, CA, pp. 3874–3879.

[7] Georges, D., Aug. 1994. Decentralized adaptive control for a water distribution system. In: Proceedings of the Third IEEE Conference on Control Applications. Glasgow, UK, pp. 1411–1416 vol.2.

[8] Giselsson, P., Doan, M. D., Keviczky, T., De Schutter, B., Rantzer, A., 2011. A distributed optimization algorithm with convergence rate O($1/k^2$) for distributed model predictive control. Tech. rep., available: `http://www.dcsc.tudelft.nl/~bdeschutter/pub/abs/12_011.html`. Submitted to Automatica.

[9] Giselsson, P., Rantzer, A., Dec. 2010. Distributed model predictive control with suboptimality and stability guarantees. In: Proceedings of the 49th Conference on Decision and Control. Atlanta, GA, pp. 7272–7277.

[10] Gomez, M., Rodellar, J., Mantecon, J. A., 2002. Predictive control method for decentralized operation of irrigation canals. Applied Mathematical Modelling 26 (11), 1039–1056.

[11] Gugercin, S., Antoulas, A. C., 2004. A survey of model reduction by balanced truncation and some new results. International Journal of Control 77 (8), 748–766.

[12] Igreja, J., Cadete, F., Lemos, J., Jun. 2011. Application of distributed model predictive control to a water delivery canal. In: Proceedings of the 19th Mediterranean Conference on Control & Automation. Corfu, Greece, pp. 682–687.

[13] Igreja, J., Lemos, J., 2009. Nonlinear model predictive control of a water distribution canal pool. In: Magni, L., Raimondo, D., Allgower, F. (Eds.), Nonlinear Model Predictive Control. Vol. 384 of Lecture Notes in Control and Information Sciences. Springer, Berlin / Heidelberg, Germany, pp. 521–529.

[14] Litrico, X., Fromion, V., 2009. Modeling and Control of Hydrosystems. Springer, London, UK.

[15] Madjidian, D., Martensson, K., Rantzer, A., 2011. A distributed power coordination scheme for fatigue load reduction in wind farms. In: Proceedings of American Control Conference. San Francisco, CA, pp. 5219–5224.

[16] Mareels, I., Weyer, E., Ooi, S. K., Cantoni, M., Li, Y., Nair, G., 2005. Systems engineering for irrigation systems: Successes and challenges. Annual Reviews in Control 29 (2), 191–204.

[17] Moore, B., 1981. Principal component analysis in linear systems: Controllability, observability, and model reduction. IEEE Transactions on Automatic Control 26 (1), 17–32.

[18] Nederkoorn, E., Schuurmans, J., Schuurmans, W., Apr. 2011. Continuous nonlinear model predictive control of a hybrid water system - Application of DNPC to a Dutch polder. In: Proceedings of the 2011 IEEE International Conference on Networking, Sensing and Control. Delft, The Netherlands, pp. 209–214.

[19] Negenborn, R., De Schutter, B., Hellendoorn, J., Apr. 2008. Multi-agent model predictive control for transportation networks: Serial versus parallel

schemes. Engineering Applications of Artificial Intelligence 21 (3), 353–366.

[20] Negenborn, R. R., van Overloop, P. J., Keviczky, T., De Schutter, B., Jun. 2009. Distributed model predictive control for irrigation canals. Networks and Heterogeneous Media 4 (2), 359–380.

[21] Nesterov, Y., 1983. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. Soviet Mathematics Doklady 27 (2), 372–376.

[22] Nesterov, Y., 1988. On an approach to the construction of optimal methods of minimization of smooth convex functions. Ekonomika i Matematicheskie Metody 24, 509–517, in Russian.

[23] Nesterov, Y., 2005. Smooth minimization of non-smooth functions. Mathematical Programming 103 (1), 127–152.

[24] Petrone, F., 2010. Model predictive control of a hydro power valley. Master's thesis, Politecnico di Milano, Italy.

[25] Qin, S. J., Badgwell, T. A., Jul. 2003. A survey of industrial model predictive control technology. Control Engineering Practice 11 (7), 733–764.

[26] Sahin, A., Morari, M., 2010. Decentralized model predictive control for a cascade of river power plants. In: Negenborn, R. R., Lukszo, Z., Hellendoorn, H. (Eds.), Intelligent Infrastructures. Vol. 42 of Intelligent Systems, Control and Automation: Science and Engineering. Springer Netherlands, pp. 463–485.

[27] Sandberg, H., Murray, R. M., 2009. Model reduction of interconnected linear systems. Optimal Control Applications and Methods 30 (3), 225–245.

[28] Savorgnan, C., Diehl, M., 2011. Control benchmark of a hydropower plant. Tech. rep., Hierarchical and Distributed Model Predictive Control for Large-Scale Systems (HD-MPC), available: http://www.ict-hd-mpc.eu/benchmarks/HD-MPC_hydropower_valley_publi

[29] Savorgnan, C., Romani, C., Kozma, A., Diehl, M., 2011. Multiple shooting for distributed systems with applications in hydro electricity production. Journal of Process Control 21 (5), 738 – 745, Special Issue on Hierarchical and Distributed Model Predictive Control.

[30] Sawadogo, S., Faye, R., Malaterre, P., Mora-Camino, F., 1998. Decentralized predictive controller for delivery canals. In: Proceedings of 1998 IEEE International Conference on Systems, Man, and Cybernetics. San Diego, CA, pp. 3880–3884 vol.4.

[31] Toh, K.-C., Yun, S., 2010. An accelerated proximal gradient algorithm for nuclear norm regularized least squares problems. Pacific Journal of Optimation 6, 615–640.

[32] Tseng, P., May 2008. On accelerated proximal gradient methods for convex-concave optimization, submitted to SIAM Journal on Optimization. Available: http://www.math.washington.edu/~tseng/papers/apgm.pdf.

[33] van Overloop, P.-J., 2006. Model Predictive Control on Open Water Systems. Ph.D. thesis, Delft University of Technology, The Netherlands.

[34] van Overloop, P.-J., Clemmens, A. J., Strand, R. J., Wagemaker, R. M. J., Bautista, E., 2010. Real-time implementation of model predictive control on Maricopa-Stanfield irrigation and drainage district's WM canal. Journal of Irrigation and Drainage Engineering 136 (11), 747–756.

[35] Vandendorpe, A., Dooren, P., 2008. Model reduction of interconnected systems. In: Schilders, W., Vorst, H., Rommes, J. (Eds.), Model Order Reduction: Theory, Research Aspects and Applications. Vol. 13 of Mathematics in Industry. Springer Berlin Heidelberg, pp. 305–321.

[36] Water Partnership Program, 2011. Wpp 2011 annual report. Http://water.worldbank.org/publications/wpp-2011-annual-report.

Appendix A.

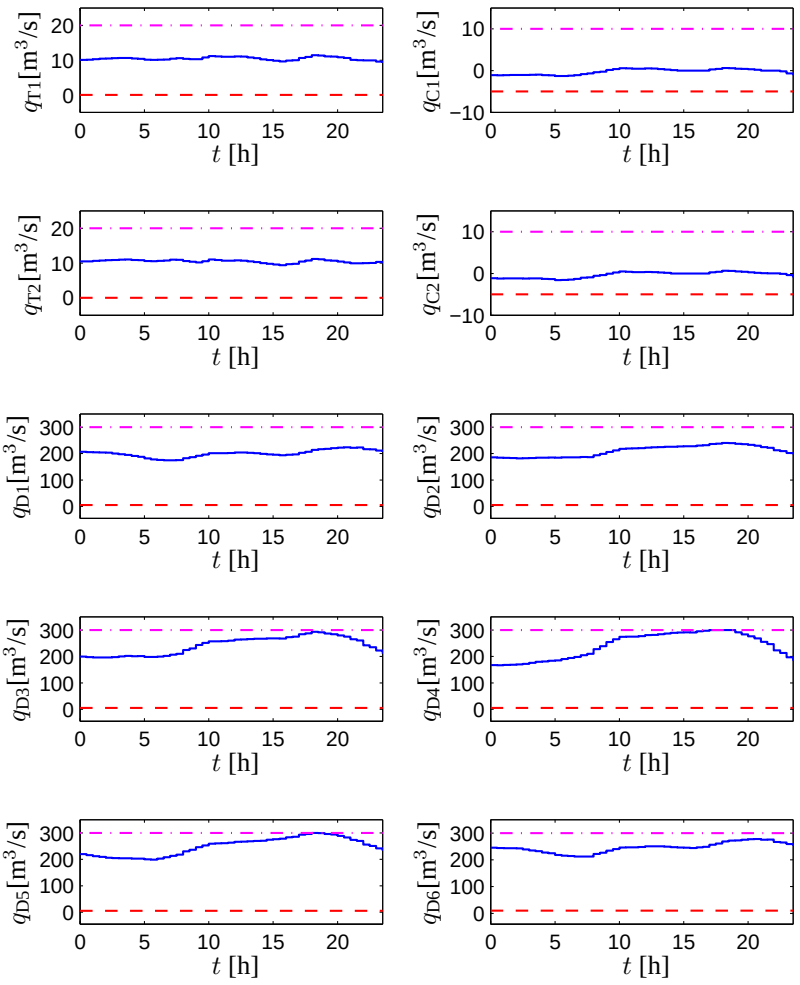


Figure A.3: Input constraint satisfaction using Algorithm 2 and power division LOC–REF–DYN. Dash-dotted lines: upper bounds, dashed lines: lower bounds.

Appendix B.

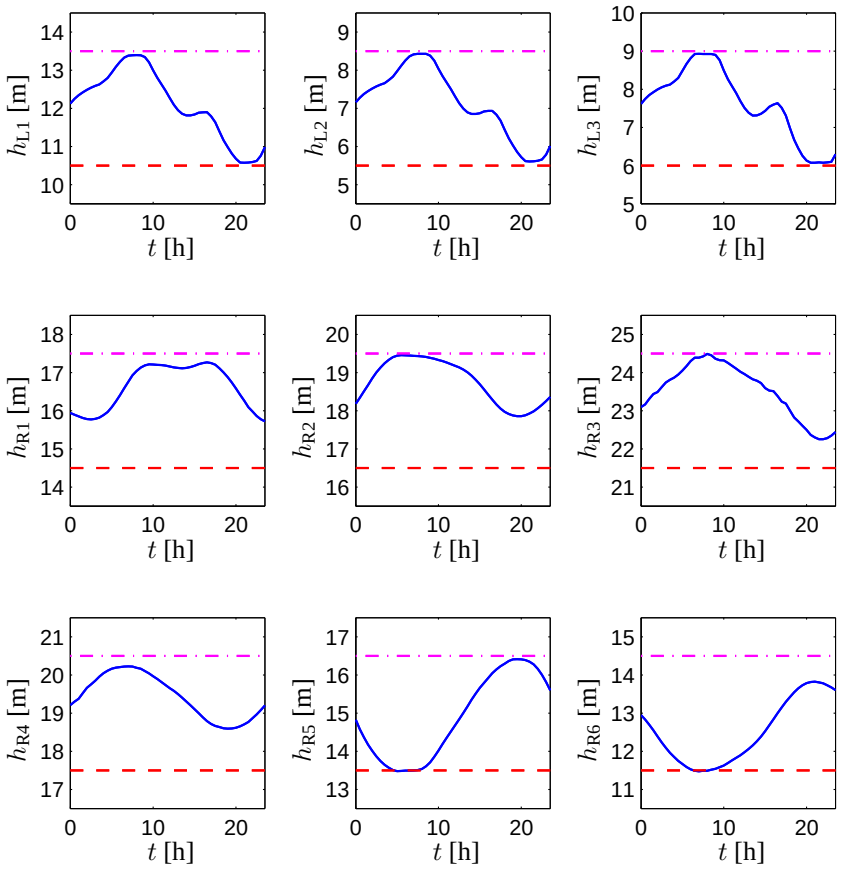


Figure B.4: Output constraint satisfaction using Algorithm 2 and power division LOC–REF–DYN. Dash-dotted lines: upper bounds, dashed lines: lower bounds.